BAB I

PENDAHULUAN

A. Latar Belakang

Perkembangan teknologi informasi yang sangat pesat telah membawa perubahan signifikan dalam berbagai aspek kehidupan, termasuk dalam bidang pendidikan dan pembelajaran. Salah satu cabang ilmu yang membutuhkan pemahaman mendalam adalah elektronika, di mana pengenalan komponen-komponen dasar menjadi fondasi penting bagi pembelajar. Namun, metode pembelajaran konvensional seringkali menghadapi kendala dalam memberikan visualisasi yang jelas dan interaktif mengenai bentuk, fungsi, dan karakteristik komponen elektronika.

Pembelajaran komponen elektronika secara tradisional biasanya mengandalkan buku teks dan gambar dua dimensi yang terbatas dalam memberikan pengalaman hands-on kepada pembelajar. Hal ini dapat menyebabkan kesulitan bagi siswa dalam memahami bentuk fisik komponen, mengidentifikasi jenis-jenis komponen, dan memahami fungsinya dalam rangkaian elektronika. Selain itu, keterbatasan akses terhadap komponen fisik di laboratorium juga menjadi hambatan dalam proses pembelajaran.

Di sisi lain, perkembangan teknologi *Augmented Reality* (AR) membuka peluang baru dalam metode pembelajaran yang lebih interaktif dan immersif. AR memungkinkan penggabungan objek virtual dengan dunia nyata secara real-time, memberikan pengalaman visual yang lebih kaya dan memungkinkan interaksi langsung dengan objek

pembelajaran. Teknologi ini, dikombinasikan dengan penetrasi perangkat *mobile* Android yang tinggi di masyarakat, dapat menjadi solusi efektif untuk mengatasi keterbatasan dalam pembelajaran komponen elektronika.

Berdasarkan permasalahan tersebut, pengembangan aplikasi pengenalan komponen elektronika dengan teknologi *Augmented Reality* berbasis Android menjadi relevan dan penting. Aplikasi ini akan memungkinkan pengguna untuk melihat visualisasi 3D komponen elektronika, berinteraksi dengan objek virtual, dan mendapatkan informasi detail tentang setiap komponen melalui perangkat mobile mereka. Pendekatan ini tidak hanya akan meningkatkan pemahaman tentang komponen elektronika tetapi juga menciptakan pengalaman pembelajaran yang lebih menarik dan efektif.

Platform Android dipilih sebagai basis pengembangan aplikasi mengingat dominasinya dalam pasar *smartphone* di Indonesia dan kemampuannya dalam menjalankan aplikasi AR dengan baik. Selain itu, penggunaan Android juga memungkinkan aksesibilitas yang lebih luas karena mayoritas pelajar telah memiliki perangkat Android.

Pengembangan aplikasi ini diharapkan dapat memberikan kontribusi signifikan dalam meningkatkan kualitas pembelajaran elektronika, khususnya dalam pengenalan komponen-komponen dasar. Dengan menggabungkan teknologi AR dan platform Android, aplikasi ini akan menjadi alat pembelajaran yang efektif, interaktif, dan mudah diakses bagi para pembelajar elektronika.

B. Rumusan Masalah

Berdasarkan latar belakang di atas maka didapatkan rumusan masalahnya adalah "Aplikasi Pengenalan Komponen Elektronika Dengan *Augmented Reality* Berbasis Android"?

C. Batasan Masalah

Adapun Batasan Masalah pada judul tersebut adalah :

1. Aplikasi ini hanya dikembangkan untuk platform Android dengan menggunakan teknologi *Augmented Reality* berbasis marker.
2. Komponen elektronika yang dapat dikenali dibatasi pada komponen dasar meliputi *resistor, kapasitor, transistor, LED, dioda, dan IC dasar*, tidak mencakup komponen elektronika kompleks atau khusus.
3. Dalam implementasi *Augmented Reality*, aplikasi menggunakan marker-based tracking untuk menampilkan visualisasi 3D komponen elektronika beserta informasi dasarnya yang meliputi nama komponen, fungsi dasar, spesifikasi umum, dan simbol skematik.
4. Aplikasi membutuhkan perangkat Android dengan kamera belakang yang berfungsi baik serta pencahayaan yang cukup untuk pendeteksian marker, dengan jarak optimal pendeteksian antara 10-50 cm.

D. Tujuan Penelitian

Berikut adalah tujuan penelitian untuk skripsi Anda:

Tujuan Penelitian:

1. Tujuan Umum:

Mengembangkan aplikasi pengenalan komponen elektronika menggunakan teknologi *Augmented Reality* berbasis Android yang dapat membantu pengguna dalam mempelajari dan memahami berbagai komponen elektronika secara interaktif dan visual.

2. Tujuan Khusus:
    a) Merancang dan membangun aplikasi berbasis Android yang dapat menampilkan visualisasi 3D komponen-komponen elektronika menggunakan teknologi *Augmented Reality*.
    b) Mengimplementasikan fitur pengenalan marker untuk memunculkan objek 3D komponen elektronika beserta informasi detailnya.
    c) Menyediakan informasi detail tentang spesifikasi, fungsi, dan karakteristik setiap komponen elektronika dalam bentuk yang mudah dipahami.
    d) Mengembangkan antarmuka pengguna (user interface) yang intuitif dan user-friendly untuk memudahkan pengguna dalam mengoperasikan aplikasi.
    e) Menguji tingkat keberhasilan aplikasi dalam menampilkan objek 3D komponen elektronika dan informasinya melalui teknologi *Augmented Reality*.
    f) Mengevaluasi tingkat kemudahan penggunaan (*usability*) aplikasi melalui pengujian dengan pengguna.

E. Manfaat Penelitian

Manfaat yang diharapkan dengan adanya Aplikasi Pengenalan Bentuk Dan Suara Hewan Berbasis *Augmented Reality* yaitu:

1. Bagi Mahasiswa
   a. Sebagai referensi pembuatan aplikasi selanjutnya berbasis *Augmented Reality*.
   b. Mengetahui cara menerapkan gambar 3 dimensi dalam sebuah marker seperti Aplikasi Pengenalan Komponen Elektronika Dengan *Augmented Reality*.
2. Bagi Kampus
   a. Sebagai referensi untuk meningkatkan pengetahuan informasi pengenalan komponen eletronik kepada siswa.
   b. Mendapatkan alternatif baru dalam memperkenalkan suara dan jenis eletronik dalam bentuk 3D.

BAB II

TINJAUAN PUSTAKA

A. Kajian Teori

1. *Augmented Reality*

Ismayani, Ani (2020), *Augmented Reality* atau AR adalah sebuah teknologi yang menggabungkan objek 2D atau 3D buatan komputer ke dalam lingkungan nyata di sekitar pengguna secara real time. Objek yang ditampilkan AR membantu pengguna dalam menghasilkan persepsi baru yang memungkinkannya berinteraksi dengan lingkungan nyata. Penggunaan AR sudah sangat umum, mulai dari game sampai dunia pendidikan dan pembelajaran. Tersedianya aplikasi gratis seperti *Unity* dan *Vuforia* memungkinkan pembuatan aplikasi berbasis AR tidak sulit dilakukan oleh siapa saja.

Teknologi AR ini dapat menyisipkan suatu informasi tertentu ke dalam dunia maya dan menampilkannya di dunia nyata dengan bantuan perlengkapan seperti webcam, komputer, HP *Android*, maupun kacamata khusus. User ataupun pengguna didalam dunia nyata tidak dapat melihat objek maya dengan mata telanjang, untuk mengidentifikasi objek dibutuhkan perantara berupa komputer dan kamera yang nantinya akan menyisipkan objek maya ke dalam dunia nyata. Metode yang dikembangkan pada *Augmented Reality* saat ini terbagi menjadi dua metode, yaitu *Marker* Based Tracking dan Markless *Augmented Reality*.

a. *Marker Augmented Reality* (*Marker* Based Tracking)

Marker biasanya merupakan ilustrasi hitam dan putih persegi dengan batas hitam tebal dan latar belakang putih. Komputer akan mengenali posisi dan orientasi *Marker* dan menciptakan dunia virtual 3D yaitu titik (0,0,0) dan tiga sumbu yaitu X, Y, dan Z. *Marker Based Tracking* ini sudah lama dikembangkan sejak 1980-an dan pada awal 1990-an mulai dikembangkan untuk penggunaan *Augmented Reality*.

b. *Markelles Augmented Reality*

Salah satu metode *Augmented Reality* yang saat ini sedang berkembang adalah metode "*Marker*less *Augmented Reality*", dengan metode ini pengguna tidak perlu lagi menggunakan sebuah *Marker* untuk menampilkan elemen-elemen digital, dengan tool yang disediakan *Qualcomm* untuk pengembangan *Augmented Reality* berbasis *mobile device*, mempermudah pengembang untuk membuat aplikasi yang *Marker*less.

2. *Unity*

Zaky, Ali (2016), *Unity* merupakan game engine yang ber-multi*platform*. *Unity* mampu di publish menjadi Standalone (.exe), berbasis web, *Android*, IoS Iphone, XBOX, dan PS3. Walaupun bisa dipublish ke berbagai *platform*. *Unity* perlu lisensi untuk dapat dipublish ke *platform* tertentu. Tetapi *Unity* menyediakan untuk free user dan bisa di *publish* dalam bentuk *Standalone* (.exe) dan web. Untuk saat ini *Unity* sedang di kembangkan berbasis AR (*Augmented Reality*). Fitur – fitur dalam *Unity* 3D adalah :

a. *Rendering*

Graphics engine yang digunakan adalah *Direct3D* (*Windows, Xbox 360*), *OpenGL* (*Mac, Windows, Linux, PS3*), OpenGL ES (*Android*, iOS), dan proprietary APIs (Wii). Ada pula kemampuan untuk bump mapping, reflection *mapping, parallax mapping, screen space ambient occlusion (SSAO), dynamic shadows using shadow maps, render – to – texture and full – screen post – processing effects.*

b. *Scripting*

Script game engine dibuat dengan Mono 2.6, sebuah implementasi open – *source dari .NET Framework. Programmer* dapat menggunakan *Unity*Script (Bahasa Terkustomisasi yang terinspirasi dari sintax ECMAScript, dalam bentuk JavaScript), C#, atau Boo (terinsipirasi dari sintax Bahasa pemrograman phyton). Dimulai dengan dirilisnya versi 3.0, *Unity* menyertakan versi MonoDevelop yang terkustomisasi untuk debug script.

c. Asset Tracking

Unity juga menyertakan Server *Unity* Asset – sebuah solusi terkontrol untuk developer game asset dan script.

d. *Platforms*

Unity support pengembangan ke berbagai *platform*. Didalam *project*, *developer* memiliki kontrol untuk mengirim keperangkat mobile, *web browser*, desktop, dan console. *Unity* juga mengijinkan spesifikasi kompresi textur dan pengaturan resolusi di setiap *platform* yang didukung.

e. Asset Store

Diluncurkan Novmeber 2010, *Unity* Asset Store adalah sebuah resource yang hadir di *Unity* editor. Asset store terdiri dari koleksi lebih dari 4,400 asset packages, beserta 3D models, texture dan materials, sistem particle, musik dan efek suara, tutorial dan project, scripting package, editor extensions dan servis online.

f. Physics

Unity juga memiliki suport built – in untuk PhysX physics engine (sejak *Unity* 3.0) dari Nvidia (sebelumnya Ageia) dengan penambahan kemampuan untuk simulasi real – time cloth pada arbitrary dan skinned meshes, thick ray cast, dan collision layers.

3. *Vuforia*

Putra, Adhitya Wibawa (2015), *Vuforia* merupakan SDK yang dikembangkan oleh Qualcomm untuk membantu pengembang dalam menciptakan aplikasi atau game yang memiliki teknologi *Augmented Reality*. Tentunya aplikasi maupun game yang dibuat dengan teknologi ini akan terlihat lebih interaktif dan hidup. Contohnya ketika pembaca mendapatkan sebuah penanda yang hanya berupa kertas dan secara tiba-tiba akan muncul objek virtual 3 dimensi ketika ponsel pintar atau tablet diarahkan ke kertas penanda tersebut.

Pengembang dapat membuat objek virtual 3 dimensi itu dapat berinteraksi dengan pengguna aplikasinya baik itu berupa game, aplikasi pembelajaran, video, aplikasi dongeng, dan masih banyak lagi. Dengan adanya *Vuforia* SDK ini akan memudahkan dan mempercepat pengembangnya dalam membuat aplikasi yang mempunyai teknologi *Augmented Reality* karena library dan fungsi-fungsi intinya

sudah dibuat oleh Qualcomm sehingga pengembang tinggal berimajinasi dan mengembangkan aplikasi menarik menggunakan SDK ini.

SDK ini sendiri memiliki berbagai fitur menarik seperti memindai objek, memindai teks, mengenali bingkai penanda, tombol virtual, mengidentifikasi permukaan objek secara pintar, memindai dengan berbasis awan, mengenali target gambar, mengenali target benda silinder, dan mengenali objek target yang telah ditetapkan. *Vuforia* SDK sendiri telah mendukung pengembangan aplikasi untuk perangkat yang berbasis iOS dan *Android*. Selain itu, *Unity* Game Engine telah didukung *Vuforia* SDK sehingga lebih mempermudah untuk mengembangkan game *Augmented Reality* langsung dengan game engine ini.

4. Jenis-Jenis Komponen Elektronika

Berikut ini merupakan fungsi dan jenis-jenis komponen elektronika dasar yang sering digunakan dalam peralatan elektronika beserta simbolnya.

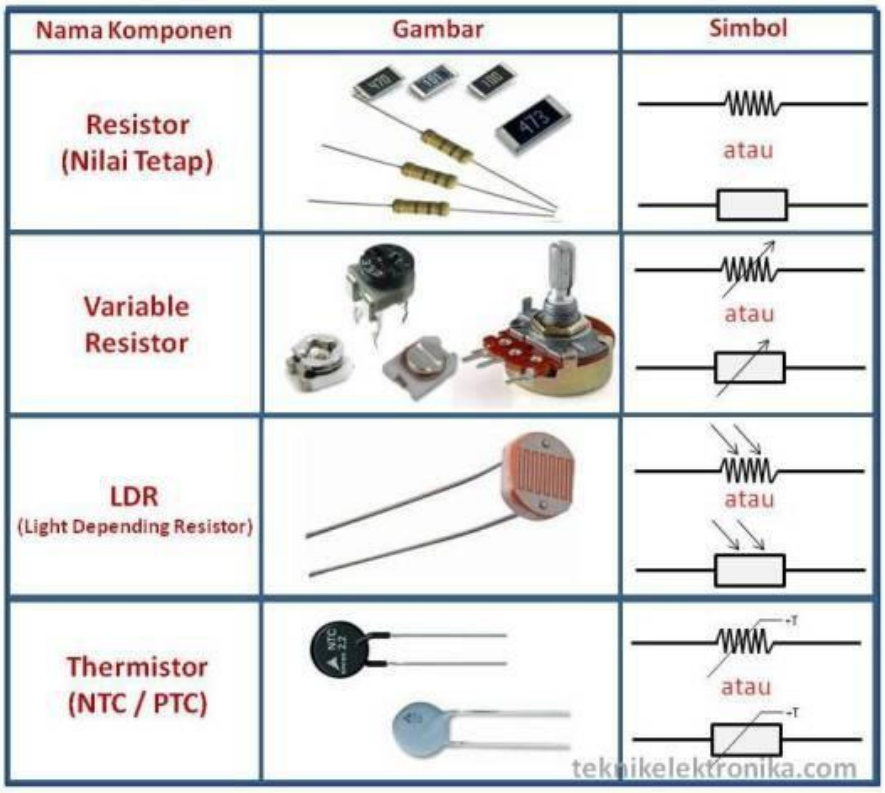
a. Resistor

Resistor atau disebut juga dengan hambatan adalah komponen elektronika pasif yang berfungsi untuk menghambat dan mengatur arus listrik dalam suatu rangkaian elektronika. Satuan nilai resistor atau hambatan adalah Ohm (Ω). Hambatan resistor sering disebut juga dengan resistansi atau *resistance.*

Jenis-jenis resistor diantaranya adalah :

1) Resistor yang nilainya tetap
2) Resistor yang nilainya dapat diatur,resistor jenis ini sering disebut juga dengan variable resistor ataupun potensiometer.

3) Resistor yang nilainya dapat berubah sesuai dengan intensitas cahaya, resistor jenis ini disebut dengan *LDR* atau *Light Dependent Resistor*
4) Resistor yang nilainya dapat berubah sesuai dengan perubahan suhu,Resistor jenis ini disebut dengan *PTC* (*Positive Temperature Coefficient*) dan *NTC (Negative Temperature Coefficient).*



Gambar 2.1 Resistor dan symbol (Sumber : teknikelektronika.com)

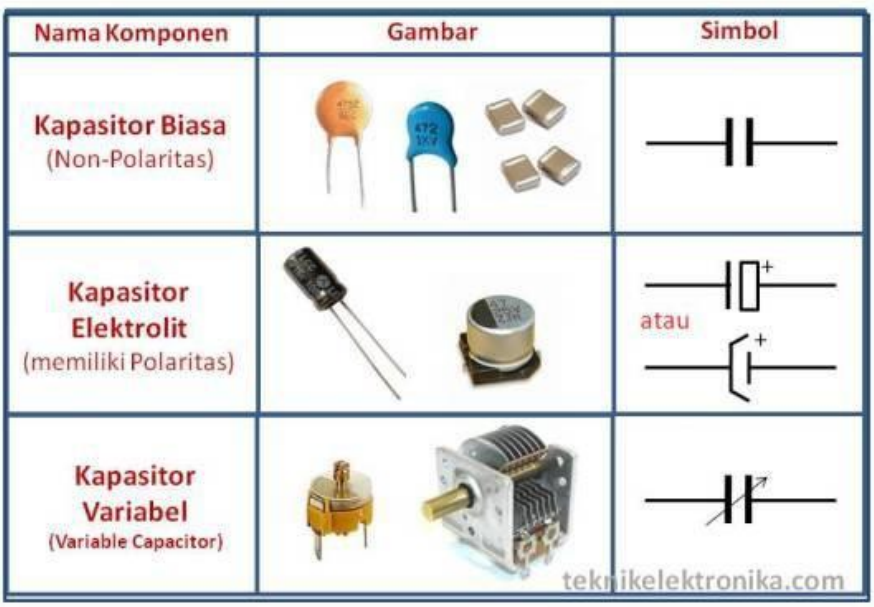
b. Kapasitor (*Capacitor*)

Kapasitor atau disebut juga dengan kondensator adalah komponen elektronika pasif yang dapat menyimpan energi atau muatan listrik dalam sementara waktu. Fungsi-fungsi kapasitor (kondensator)

diantaranya adalah dapat memilih gelombang radio pada rangkaian tuner, sebagai perata arus pada *rectifier* dan juga sebagai *filter* di dalam rangkaian power supply (Catu Daya). Satuan nilai untuk kapasitor (*Kondensator*) adalah *Farad* (F)

Jenis-jenis kapasitor diantaranya adalah :

1) Kapasitor yang nilainya tetap dan tidak ber-polaritas. Jika didasarkan pada bahan pembuatannya maka kapasitor yang nilainya tetap terdiri dari Kapasitor Kertas, Kapasitor *Mika*, Kapasitor *Polyster* dan Kapasitor Keramik.
2) Kapasitor yang nilainya tetap tetapi memiliki polaritas positif dan negatif, kapasitor tersebut adalah kapasitor elektrolit atau *electrolyte condensator (ELCO)* dan kapasitor tantalum
3) Kapasitor yang nilainya dapat diatur, kapasitor jenis ini sering disebut dengan *Variable Capasitor*. (Teknikelektronika.com)



Gambar 2.2 Kapasitor *dan symbol (Sumber : Teknikelektronika.com)*

c. Induktor (*Inductor*)

Induktor atau disebut juga dengan *Coil* (Kumparan) adalah komponen elektronika pasif yang berfungsi sebagai pengatur frekuensi, filter dan juga sebagai alat kopel (penyambung). Induktor atau *Coil* banyak ditemukan pada peralatan atau rangkaian elektronika yang berkaitan dengan frekuensi seperti *Tuner* untuk pesawat radio. Satuan Induktansi untuk Induktor adalah *Henry* (H).

Jenis-jenis Induktor diantaranya adalah :

1. Induktor yang nilainya tetap
2. Induktor yang nilainya dapat diatur atau sering disebut dengan Coil



Gambar 2.3 *Induktor dan symbol (Sumber : teknikelektronika.com)*

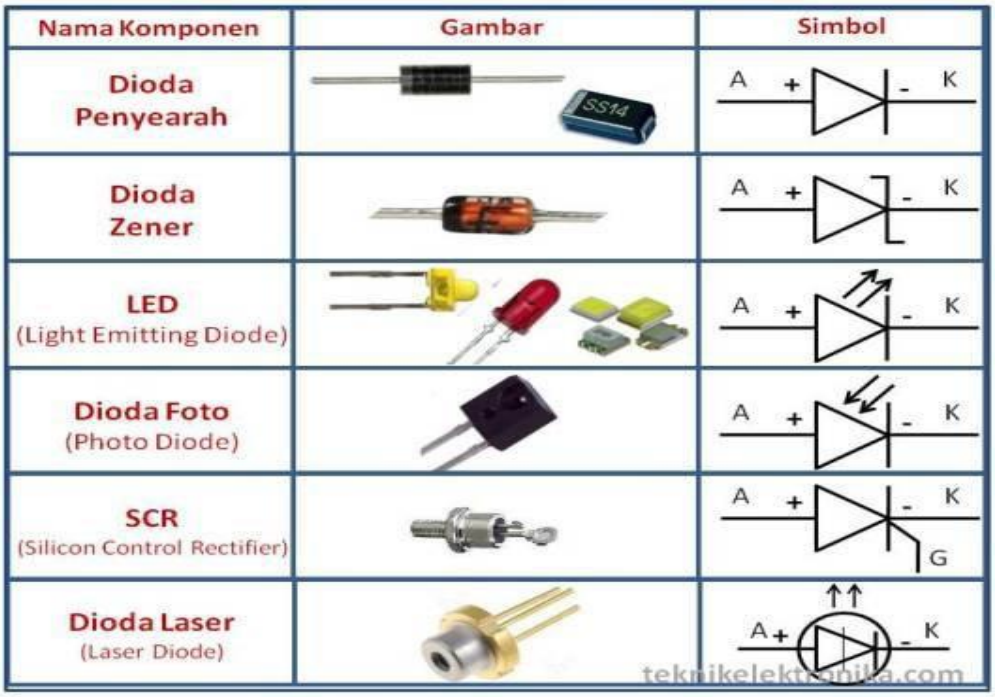
d. Dioda (*Diode*)

Diode adalah komponen elektronika aktif yang berfungsi untuk menghantarkan arus listrik ke satu arah dan menghambat arus listrik dari arah sebaliknya. *Diode* terdiri dari 2 elektroda yaitu *anoda* dan *katoda*.

Berdasarkan Fungsi Dioda terdiri dari :

1) Dioda biasa atau dioda penyearah yang umumnya terbuat dari silikon dan berfungsi sebagai penyearah arus bolak balik (*AC*) ke arus searah (*DC*).
2) Dioda zener (*Zener Diode*) yang berfungsi sebagai pengamanan rangkaian setelah tegangan yang ditentukan oleh dioda *zener* yang bersangkutan. tegangan tersebut sering disebut dengan tegangan *zener*.
3) *LED (Light Emitting Diode)* atau *diode* emisi cahaya yaitu dioda yang dapat memancarkan cahaya *monokromatik*.
4) Dioda foto (*photo diode*) yaitu dioda yang peka dengan cahaya sehingga sering digunakan sebagai Sensor.
5) Dioda *schottky* (*SCR atau Silicon Control Rectifier*) adalah dioda yang berfungsi sebagai pengendali.
6) Dioda laser *(laser diode*) yaitu dioda yang dapat memancar cahaya laser.

   Dioda Laser sering disingkat dengan LD. (Teknikelektronika.com)



Gambar 2.4 *Dioda dan symbol (Sumber : teknikelektronika.com)*

e. Transistor

Transistor merupakan komponen elektronika aktif yang memiliki banyak fungsi dan merupakan komponen yang memegang peranan yang sangat penting dalam dunia elektronik modern ini. Beberapa fungsi transistor diantaranya adalah sebagai penguat arus, sebagai switch (Pemutus dan penghubun g), stabilitasi tegangan, modulasi sinyal, penyearah dan lain sebagainya. Transistor terdiri dari 3 terminal (kaki) yaitu *base*/basis (B), *emitor* (E) dan *collector*/kolektor (C). Berdasarkan strukturnya, transistor terdiri dari 2 tipe struktur yaitu *PNP* dan *NPN*. *UJT* (*Uni Junction Transistor*), *FET (Field Effect Transistor*) dan *MOSFET* (*Metal Oxide Semiconductor FET*) juga merupakan keluarga dari Transistor. (Teknikelektronika.com)



Gambar 2.5 *Transistor dan symbol (Sumber : teknikelektronika.com)*

f. IC (*Integrated Circuit*)

IC (*integrated circuit*) adalah komponen elektronika aktif yang terdiri dari gabungan ratusan bahkan jutaan transistor, resistor dan komponen lainnya yang diintegrasi menjadi sebuah rangkaian elektronika dalam sebuah kemasan kecil. Bentuk IC (*Integrated Circuit*) juga bermacam-macam, mulai dari yang berkaki 3 (tiga) hingga ratusan kaki (terminal). Fungsi IC juga beraneka ragam, mulai dari penguat, *Switching*, pengontrol hingga media penyimpanan.



Gambar 2.6 *IC (Integrated Circuit) dan symbol (Sumber : teknikelektronika.com)*

g. Saklar (*Switch*)

Saklar adalah komponen yang digunakan untuk menghubungkan dan memutuskan aliran listrik. Dalam rangkaian elektronika, saklar sering digunakan sebagai ON/OFF dalam peralatan elektronika. (Teknikelektronika.com)



Gambar 2.7 *Saklar (Switch) dan symbol (Sumber : teknikelektronika.com*)

5. Bahasa Pemrograman C#

Filus, Teo (2017), C# atau yang dibaca C sharp adalah bahasa pemrograman sederhana yang digunakan untuk tujuan umum, dalam artian bahasa pemrograman ini dapat digunakan untuk berbagai fungsi misalnya untuk pemrograman server-side pada website, membangun aplikasi desktop ataupun mobile, pemrograman game dan sebagainya. Selain itu C# juga bahasa pemrograman yang berorientasi objek, jadi C# juga mengusung konsep objek seperti inheritance, class, polymorphism dan encapsulation. Dalam prakteknya C# sangat bergantung dengan framework yang disebut .NET Framework, framework inilah yang nanti digunakan untuk mengcompile dan menjalankan kode C#.

C# dikembangkan oleh Microsoft dengan merekrut Anders Helsberg. Tujuan dibangunnya C# adalah sebagai bahasa pemrograman utama dalam lingkungan .NET Framework (lihat C#). Banyak pihak juga yang menganggap bahwa Java dengan C# saling bersaing, bahkan ada juga yang menyatakan jika pernah belajar Java maka belajar C# akan sangat mudah dan begitu juga sebaliknya. Anggapan tersebut sebenarnya tidak salah karena perlu diketahui sebelum adanya C# Microsoft mengembangkan J++ dengan maksud mencoba membuat Java agar berjalan pada *platform* Windows, karena adanya masalah dari pihak luar maka Microsoft menghentikan proyek J++ dan beralih untuk mengembangkan bahasa baru yaitu C#.

Dalam mengembangkan aplikasi ataupun fungsi-fungsi lain yang berbasis C#, kita perlu IDE (Integrated Development Environment) untuk membantu pekerjaan kita. Di Windows kita dapat menggunakan Visual Studio yang memiliki fitur yang sangat lengkap. Akan tetapi tidak semua dari kita memakai OS (Operating System) Windows dan memiliki kapasitas harddisk yang memadai karena perlu diketahui Visual Studio ini sangat memakan tempat di harddisk kita. Karena itu ada IDE alternatif Visual Studio yaitu Xamarin Studio yang bisa dipakai lintas *platform* yang fiturnya menurut saya tidak kalah hebatnya dengan Visual Studio dan lebih dari cukup untuk membangun aplikasi C#. Selain itu ada juga tool seperti SharpDevelop namun sepertinya hanya bisa untuk *platform* Windows. Oh iya bagi pembaca yang memakai OS selain windows dapat memakai Mono untuk pengganti .NET Framework. Dalam pemrograman C# (mungkin juga berlaku untuk beberapa pemrograman lainnya) memiliki 5 struktur dasar yang harus diingat yaitu:

a. Resource atau library

Struktur pertama ini merupakan pendefinisian library apa yang harus ada pada program kita atau library apa yang kita impor.

b. Namespace

Struktur kedua ini adalah nama dari project kita.

c. Nama Class

Struktur ketiga ini berbicara tentang apa nama dari Class yang kita buat dan bisa juga langsung diberi penanda seperti Main Class yang menandakan bahwa Class tersebut Class utama.

d. Deklarasi Method

Struktur keempat ini merupakan pendeklarasian method sebagai awalan untuk menjalankan method atau perintah yang ada di dalamnya, jika didefinisikan dengan "Main" maka method tersebut yang dijalankan pertama kali oleh compiler.

e. Method atau Command

Struktur kelima adalah method atau perintah yang kita berikan untuk di eksekusi oleh compiler.

6. *Android*

Komputer, Wahana (2013), *Android* adalah sebuah system operasi Mobile yang berbasiskan pada versi modifikasi dari linux. Pertama kali system operasi ini dikembangkan oleh perusahaan *Android* .inc. Nama perusahaan inilah yang pada akhirnya digunakan sebagai nama proyek system operasi mobile tersebut yaitu system operasi *Android*.

Pada tahun 2005, sebagai bagian dari strategi untuk memasuki pasar Mobile, Google membeli *Android* dan mengambil alih proses pengembangannya sekaligus team developer *Android*.Google menginginkan *Android* untuk menjadi system operasi yang open source dan gratis, kebanyakan code *Android* dirilis dibawah lisensi open source Apache yang berarti setiap orang bebas untuk menggunakan dan mengunduh source code *Android* secara penuh. Fitur – Fitur *Android* :

a. Storage, mendukung SQL Lite. SQL Lite adalah sebuah database relational Lite (versi ringan) yang digunakan untuk menyimpan data.
b. Konektivitas, mendukung koneksi GSM (EDGE), IDEN, SDMA, SVDO, UMTS, Bluetooth, WIFI, LTE, dan Wimax.
c. Messaging, mendukung SMS, dan MMS.
d. Webrowser, yang digunakan adalah browser berbasis open source

7. *UML (Unified Modelling Language)*

Sora (2018), *Unified Modelling Language (UML)* adalah sebuah "bahasa" yg telah menjadi standar dalamindustri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. *UML*menawarkan sebuah standar untuk merancang model sebuah sistem.

UML adalah sekumpulan alat yang digunakan untuk melakukan abstraksiterhadap sebuah sistem atau perangkat lunak berbasis objek. *UML* merupakan singkatan dari *Unified Modeling Language*. *UML* juga menjadi salah satu cara untuk mempermudah pengembangan aplikasi yang berkelanjutan. Aplikasi atau sistem yang tidak terdokumentasi biasanya dapat menghambat

pengembangan karena *developer* harus melakukan penelusuran dan mempelajari kode program *UML* juga dapat menjadi alat bantu untuk transfer ilmu tentang sistem atau aplikasi yang akan dikembangkan dari satu *developer* ke *developer* lainya. Tidak hanya antar *developer* terhadap orang bisnis dan siapapun dapat memahami sebuah sistem dengan adanya *UML*.

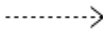
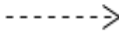
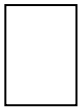
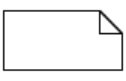
UML diciptakan oleh *Object Management Group* yang diawali dengan versi 1.0 pada Januari 1997. Dalam pengembangan berorientasi objek ada beberapa prinsip yang harus dikenal:*Object, Class, Abstraction, Encapsulation, Inheritance dan Polymorphism*. Dalam *UML* sendiri terdapat beberapa diagram yaitu :

a. *Use Case Diagram*

Use Case diagram menggambarkan *fungsionalitas* yang diharapkan dari sebuah sistem. Yang ditekankan adalah "apa" yang diperbuat sistem, dan bukan "bagaimana". Sebuah *Use Case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use Case* merupakan sebuah pekerjaan tertentu, misalnya login ke sistem, meng-create sebuah daftar belanja, dan sebagainya. Seorang/sebuah aktor adalah sebuah *entitas* manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu. Adapun simbol-simbol *Use Case Diagram* antara lain :

Tabel 2. 1. Simbol *Use Case* Diagram

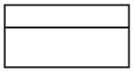
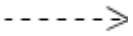
No	Gambar	Nama	Keterangan
1		*Actor*	Menspesifikasikan himpuan peran yang pengguna *Main*kan ketika berinteraksi dengan *Use Case*.

No	Gambar	Nama	Keterangan
2		*Dependency*	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri *(independent)* akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (*independent*).
3		*Generalization*	Hubungan dimana objek anak (*descendent*) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*).
4		*Include*	Menspesifikasikan bahwa *Use Case* sumber secara *eksplisit*.
5		*Extend*	Menspesifikasikan bahwa *Use Case* target memperluas perilaku dari *Use Case* sumber pada suatu titik yang diberikan.
6		*Association*	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		*System*	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		*Use Case*	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
9		*Collaboration*	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan prilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		*Note*	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

b. *Class Diagram*

Adapun simbol-simbol *Class Diagram* antara lain :

Tabel 2. 2. Simbol *Class* Diagram

No	Gambar	Nama	Keterangan
1		*Generalization*	Hubungan dimana objek anak *(descendent)* berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*).
2		*Nary Association*	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		*Class*	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		*Collaboration*	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
5		*Realization*	Operasi yang benar-benar dilakukan oleh suatu objek.
6		*Dependency*	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri *(independent)* akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7		*Association*	Apa yang menghubungkan antara objek satu dengan objek lainnya

c. *Sequence Diagram*

Adapun simbol-simbol *Sequence* Diagram antara lain :

Tabel 2. 3. Simbol *Sequence* Diagram

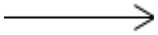
No	Gambar	Nama	Keterangan
1		*LifeLine*	Objek *entity*, antarmuka yang saling berinteraksi.

2		*Message*	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
3		*Message*	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

d. *StateChart Diagram*

Adapun simbol-simbol *StateChart Diagram* antara lain :

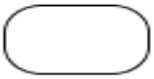
Tabel 2. 4. Simbol *StateChart* Diagram

No	Gambar	Nama	Keterangan
1		*State*	Nilai atribut dan nilai link pada suatu waktu tertentu, yang dimiliki oleh suatu objek.
2		*Initial Pseudo State*	Bagaimana objek dibentuk atau diawali
3		*Final State*	Bagaimana objek dibentuk dan dihancurkan
4		*Transition*	Sebuah kejadian yang memicu sebuah state objek dengan cara memperbaharui satu atau lebih nilai atributnya
5		*Association*	Apa yang menghubungkan antara objek satu dengan objek lainnya.
6		*Node*	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

e. *Activity Diagram*

Adapun simbol-simbol *Activity Diagram* antara lain :

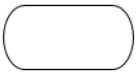
Tabel 2. 5. Simbol *Activity* Diagram

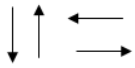
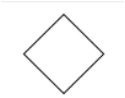
No	Gambar	Nama	Keterangan
1		*Activity*	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		*Action*	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		*Initial Node*	Bagaimana objek dibentuk atau diawali.
4		*Activity Final Node*	Bagaimana objek dibentuk dan dihancurkan
5		*Fork Node*	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

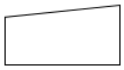
8. *Flowchart*

Rahmad Arifianto, (2017). *Flowchart* adalah adalah suatu bagan dengan simbol-simbol tertentu yang menggambarkan urutan proses secara mendetail dan hubungan antara suatu proses (instruksi) dengan proses lainnya dalam suatu program. Berikut ini adalah beberapa simbol yang digunakan dalam menggambar suatu *Flowchart*:

Tabel 2. 6. Simbol *Flowchart*

Simbol	Nama	Fungsi
	***Terminal Point Symbol***	*Terminal Menu*njukkan permulaan (start) atau akhir (stop) dari suatu proses.

Simbol	Nama	Fungsi
	***Flow Direction Symbol***	Adalah simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan simbol yang lain (*connecting line*). Simbol ini juga berfungsi untuk *Menu*njukkan garis alir dari proses.
	***Processing Symbol***	Digunakan untuk *Menu*njukkan kegiatan yang dilakukan oleh komputer. Pada bidang industri (proses produksi barang), simbol ini menggambarkan kegiatan inspeksi atau yang biasa dikenal dengan simbol inspeksi
	***Decision Symbol***	Merupakan simbol yang digunakan untuk memilih proses atau keputusan berdasarkan kondisi yang ada. Simbol ini biasanya ditemui pada *Flowchart* program.
	**I*nput-Output***	Masuk menunjukkan proses *input-output* yang terjadi tanpa bergantung dari jenis peralatannya.
	***Predefined Process***	*Terdefinisi* merupakan simbol yang digunakan untuk *Menu*njukkan pelaksanaan suatu bagian prosedur (*sub*-proses).
	***Connector (On-page)***	Simbol ini fungsinya adalah untuk menyederhanakan hubungan antar simbol yang letaknya berjauhan atau rumit bila dihubungkan dengan garis dalam satu halaman
	***Connector (Off-page)***	Sama seperti *on-page connector*, hanya saya simbol ini digunakan untuk menghubungkan simbol dalam halaman berbeda. label dari simbol ini dapat menggunakan huruf atau angka
	***Preparation Symbol***	Merupakan simbol yang digunakan untuk mempersiapkan penyimpanan di dalam storage.

Simbol	Nama	Fungsi
	***Manual Input Symbol***	Digunakan untuk menunjukkan input data secara manual menggunakan online keyboard.
	***Manual Operation Symbol***	Yang digunakan untuk menunjukkan kegiatan/proses yang tidak dilakukan oleh komputer.
	***Document Symbol***	Jika Anda menemukan simbol ini artinya input berasal dari dokumen dalam bentuk kertas, atau output yang perlu dicetak di atas kertas.
	***Multiple Documents***	Sama seperti *document symbol* hanya saja dokumen yang digunakan lebih dari satu dalam simbol ini
	***Display Symbol***	Adalah simbol yang menyatakan penggunaan peralatan output, seperti layar monitor, printer, plotter dan lain sebagainya
	***Delay Symbol***	Sesuai dengan namanya digunakan untuk *Menu*njukkan proses delay (*Menu*nggu) yang perlu dilakukan. Seperti *Menu*nggu surat untuk diarsipkan

B. Kajian Hasil Penelitian Terdahulu

Melihat dari sejumlah judul dan tema yang berkaitan dengan penelitian yang dilakukan, diperoleh persamaan dan perbedaan dari penelitian yang akan dilakukan. Tujuannya membuktikan bahwa penulisan penelitian asli dan bukan duplikasi dari penelitian lain, seperti berikut ini :

1. Nadira, Sujaini dan Pratiwi (2016) dalam penelitianya yang berjudul "Implementasi Augmented Reality Pada Brosur Teknik Informatika Universitas Tanjungpura Menggunakan Metode *Marker* " saat ini brosur

Teknik Informatika Universitas Tanjungpura masih berisi gambar-gambar dari kampus itu sendiri, tetapi gambar-gambar tersebut masih kurang memberikan informasi yang jelas sehingga calon mahasiswa sulit untuk membayangkan gambaran dari kampus Teknik Informatika Universitas Tanjungpura dan akhirnya calon mahasiswa tidak terlalu tertarik untuk memilih Teknik Informatika Universitas Tanjungpura. Manfaat dari penerapan *Augmented Reality* pada brosur Teknik Informatika Universitas Tanjungpura yaitu dapat memberikan gambaran yang jelas karena dilengkapi dengan gambar bangunan kampus dalam bentuk 3 dimensi. Dalam penelitian ini menggunakan metode *Marker* Based Tracking. Hasil yang didapat dari penelitian ini adalah peneliti menguji aplikasi yang dibuat dan hasilnya berjalan lancar pada handphone *Android* dan pengguna menyatakan setuju bahwa aplikasi Augmented Relaity brosur dapat memperjelas informasi pada brosur Teknik Informatika Universitas Tanjungpura.

2. Hermawan dan Hariadi (2015) dalam penelitianya yang berjudul "Pemanfaatan *Augmented Reality* Sebagai Media Informasi Kampus Menggunakan Brosur" Manfaat dari penerapan *Augmented Reality* pada brosur kampus yaitu pembaca dapat melihat bagian brosur Augmented Relaity dengan detail serta lingkungan pada brosur juga akan terasa lebih hidup dengan adanya animasi pendukung seperti gedung, pohon, dan lain sebagainya. Metode penelitian yang akan digunakan dalam pembuatan brosur berbasis *Augmented Reality* yaitu terdiri dari empat proses yaitu perencanaan, pengumpulan data, analisis dan dokumentasi hasil yang telah

dibuat. Hasil yang didapat dari penelitian ini adalah aplikasi berjalan lancar dan telah diuji menggunakan komputer dan dapat menampilkan objek 3 dimensi pada brosur yang dapat memberikan informasi yang belum ada pada brosur sebelumnya.

Berdasarkan beberapa penelitian yang telah dilakukan maka dilakukan penelitian lebih lanjut untuk dapat membuat sistem aplikasi *Android* virtual jam tangan berbasis *Augmented Reality* dengan *Marker* based tracking *Vuforia.* Pada penelitian ini terdapat perbedaan dengan kajian terdahulu, yaitu objek penelitian, tahun penelitian dan pembahasan tentang penerapan *Augmented Reality* untuk menampilakan dan mengenal suara hewan. Bahasa pemograman yang digunakan adalah C# dan *Marker* based menggunakan *Vuforia*.

C. Black Box

Black box testing adalah metode pengujian perangkat lunak di mana penguji mengevaluasi fungsionalitas aplikasi tanpa melihat kode sumber atau struktur internalnya. Fokus utama dari pengujian ini adalah pada input dan output sistem. Penguji memberikan berbagai input untuk memastikan bahwa aplikasi menghasilkan hasil yang diharapkan sesuai dengan spesifikasi.

a. Ciri-ciri Black Box Testing

1) **Fokus pada Fungsionalitas**: Menguji apakah aplikasi berfungsi seperti yang diharapkan.
2) **Tanpa Pengetahuan Kode**: Penguji tidak perlu memahami bagaimana sistem bekerja di balik layar.

3) **Metode Pengujian**: Meliputi pengujian fungsional, pengujian regresi, dan pengujian pengguna.

b. Kelebihan Black Box Testing

1) **Mendapatkan Perspektif Pengguna**: Memungkinkan pengujian berdasarkan bagaimana pengguna akan berinteraksi dengan aplikasi.
2) **Dapat Mengidentifikasi Kesalahan Fungsional**: Efektif dalam menemukan bug yang mempengaruhi pengguna akhir.

c. Kekurangan Black Box Testing

1) **Tidak Mengidentifikasi Masalah *Intern***: Tidak dapat menemukan masalah terkait dengan struktur atau kode internal.
2) **Memerlukan Spesifikasi yang Jelas**: Bergantung pada dokumen spesifikasi yang baik untuk pengujian yang efektif.

Secara keseluruhan, black box testing adalah teknik penting dalam pengujian perangkat lunak yang membantu memastikan kualitas dan kinerja aplikasi dari sudut pandang pengguna.

D. White Box

White box testing adalah metode pengujian perangkat lunak yang melibatkan pengujian internal dari aplikasi dengan memeriksa kode sumber, struktur, dan algoritma yang digunakan. Dalam pengujian ini, penguji memiliki pengetahuan yang mendalam tentang kode dan dapat melihat bagaimana sistem berfungsi dari dalam.

c. Ciri-ciri White Box Testing

1) **Fokus pada Kode**: Penguji menganalisis dan memahami logika dan struktur kode sumber.

2) **Pengujian Fungsional dan Non-Fungsional:** Selain menguji fungsionalitas, juga dapat menguji aspek non-fungsional seperti keamanan dan performa.
3) **Penggunaan Alat**: Sering kali melibatkan penggunaan alat otomasi untuk memeriksa kode dan menjalankan pengujian.

d. Kelebihan White box test

1) **Identifikasi Masalah Internal**: Mampu menemukan bug yang mungkin tidak terlihat dalam black box testing, seperti kesalahan logika atau kesalahan dalam algoritma.
2) **Optimasi Kode**: Dapat membantu meningkatkan kualitas dan efisiensi kode dengan mengidentifikasi bagian yang perlu perbaikan.

e. Kekurangan White box test

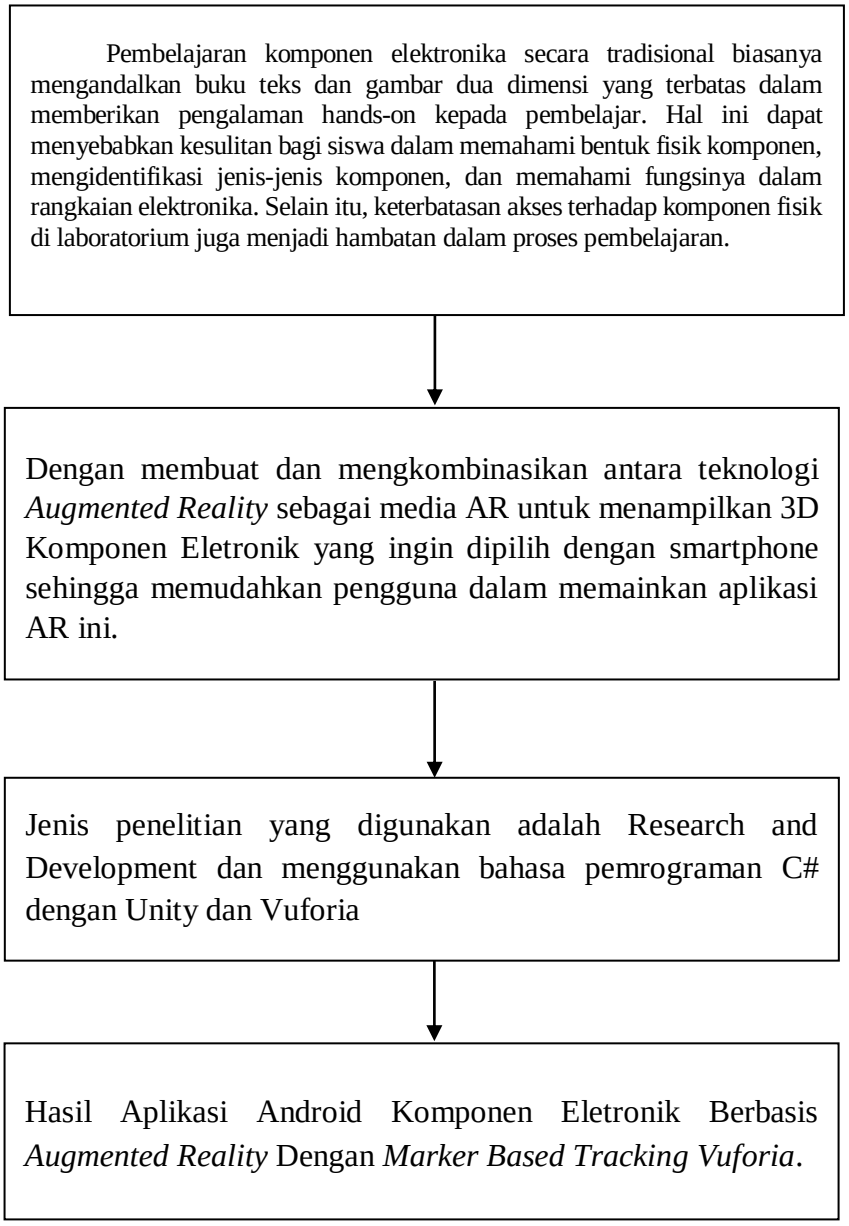
1) **Memerlukan Pengetahuan Mendalam**: Penguji harus memiliki pemahaman yang baik tentang bahasa pemrograman dan struktur kode.
2) **Waktu dan Biaya**: Proses pengujian bisa lebih memakan waktu dan biaya, terutama untuk aplikasi besar dengan banyak kode.

f. Jenis Pengujian White box test

1) **Unit Testing**: Menguji bagian terkecil dari aplikasi secara terpisah.
2) **Integration Testing**: Menguji interaksi antara modul-modul yang berbeda dalam aplikasi.
3) White box testing sangat efektif untuk memastikan kualitas dan keandalan perangkat lunak dari sudut pandang internal, sehingga sering digunakan dalam pengembangan perangkat lunak yang kompleks.

E. Kerangka Pikir

Untuk lebih memperjelas kerangka pikir maka digambarkan dalam bentuk diagram sebagai berikut :



Gambar 2. 1. Kerangka Pikir

BAB III

METODE PENELITIAN

A. Lokasi Dan Waktu Penelitian

Proses penelitian dilakukan di SMK Muhammadiyah Parepare. Waktu yang dipergunakan untuk penelitian ini 2 bulan.

Tabel 3. 1. Jadwal Pelaksanaan Penelitian

No.	Uraian Kegiatan	Bulan 2024-2025		
		November	Desember	Januari
1	Studi Literatur			
2	Pembuatan Aplikasi			
3	Pengujian Aplikasi			
4	Hasil Pengujian			

B. Jenis Penelitian

Jenis penelitian yang digunakan adalah *Research and Development*, yaitu penelitian yang digunakan untuk menghasilkan aplikasi tertentu, dan menguji keefektifan aplikasi tersebut. Penelitian yang akan mengimplementasi augmented reality pada aplikasi pengenalan hewan, langkah-langkah dari pembuatan aplikasi ini biasanya disebut dengan siklus *Research and Development* yang terdiri dari beberapa langkah diantaranya menganalisis aplikasi yang akan dikembangkan, proses pengembangan, pengujian produk hingga tahap revisi dan evaluasi untuk memperbaiki kelemahan aplikasi.

C. Metode Pengumpulan Data

Pengumpulan data dilakukan untuk mengumpulkan seluruh informasi yang terkait dan mendukung pelaksanaan penelitian penelitian adalah kajian pustaka, metode pengumpulan data dengan cara mencari buku yang ada di perpustakaan atau mencari di internet untuk mendapatkan referensi yang serupa dengan penelitian.

D. Alat Dan Bahan Penelitian

Dalam pelaksanaan peneltian ini digunakan beberapa alat dan bahan yang terdiri:

1. Alat penelitian

a. Personal Computer dengan spesifikasi :

1) Processor : Intel Core i3 gen 7

2) Memory : 4 GB

3) HDD : 1 TB

Perangkat lunak (*software*) yang digunakan untuk membangun aplikasi ini dapat dilihat pada spesifikasi seperti Windows 10 x64, *Unity* dan Vuforia.

2. Bahan penelitian

Bahan penelitian berupa asset 3D Hewan dan marker vuforia.

E. Tahap Penelitian

Tahap penelitian dilakukan dengan cara :

1. Persiapan Penelitian

Pada tahapan ini peneliti melakukan persiapan penelitian. Persiapan penelitian yang dimaksud adalah menyiapkan buku-buku, artikel tentang topik penelitian serta *software* yang digunakan selama penelitian.

2. Pengumpulan Data

Pada tahap ini peneliti melakukan kajian pustakan untuk pencatatan dan pengamatan pembelajaran.

3. Analisis

Pada tahap analisis, peneliti melakukan analisa terhadap sistem dan data yang ada kemudian merumuskan masalah yang menjadi pokok penelitian sehingga dapat dibuat alternatif pemecahan masalah.

4. Perancangan

Peneliti kemudian merancang aplikasi pengenalan hewan dengan mengimplementasikan augmented reality berbasis *Android* yang ingin dibuat.

5. Pengujian

Setelah melakukan perancangan, peneliti kemudian menguji hasil perancangan aplikasi. Jika hasil perancangan terdapat kekurangan atau kelemahan maka kembali ke tahap analisis.

6. Implementasi

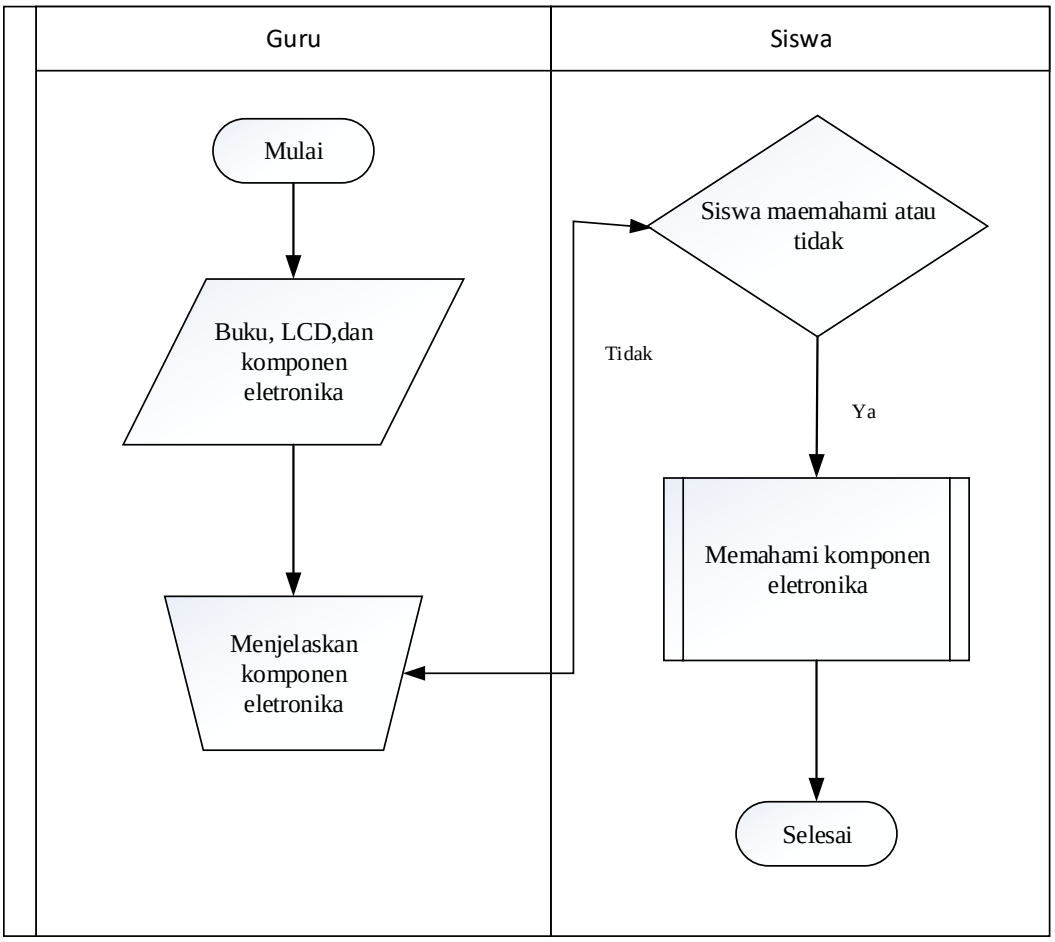
Setelah pada perancangan tidak terdapat kekurangan maka aplikasi siap untuk di gunakan.

F. Metode Pengujian

Untuk melakukan pengujian dari sistem yang dibangun yaitu penerapan menggunakan Unity dengan bahasa pemrograman C# dibuat sebuah mekanisme pengujian yang tercantum dalam rencana pengujian yang di golongkan oleh jenis pengguna. Pengujian dilakukan dengan menguji setiap proses yang ada pada sistem. Pengujian ini dilakukan secara *WhiteBox*, yaitu pengujian terhadap cara kerja perangkat lunak itu sendiri yaitu prosedur programnya (*basis path*) atau proses *looping* (pengulangan). Dan juga *BlackBox*, yaitu pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak.

G. Desain Sistem

a. Analisis Sistem yang sedang Berjalan



Penjelasan dari gambar adalah dimana setiap mahasiswa menerima pelajaran tentang pengenalan komponen elektronika dengan mendengarkan penjelasan dari guru mata pelajaran. Selain itu juga dengan melihat komponen secara langsung. Proses pembelajaran seperti ini cenderung menurunkan semangat belajar dari siswa SMK Khusunya SMK Muhammadiyah Parepare yang kemudian berdampak pada tidak tercapainya kompetensi yang diinginkan. Oleh karena itu, perancangan aplikasi pengenalan komponen elektronika dengan augmented reality berbasis

android ini dapat memberikan alternatif pembelajaran yang lebih efektif, menarik, dan interaktif yang dapat mendukung kegiatan pembelajaran.

b. Analisis Sistem yang diusulkan

Analisis sistem merupakan penguraian dari suatu sistem yang utuh ke dalam bagian-bagian komponennya untuk mengidentifikasi dan mengevaluasi permasalahan.Bagian analisis terdiri dari analisis masalah, analisis kebutuhan dan analisis kelemahan.

1. Analisis Masalah

Aplikasi pengenalan komponen elektronika dengan *augmented reality* berbasis android adalah aplikasi yang digunakan sebagai salah satu alternatif media pembelajaran komponen elektronika yang lebih menarik, interaktif dan inovatif sehingga model pembelajaran lebih variatif lagi dan di harapkan dapat memberikan semangat baru bagi peserta didik.

2. Analisis Kebutuhan
    a. Kebutuhan *Non* Fungsional

Analisis kebutuhan non fungsional menggambarkan kebutuhan sistem yang menitik beratkan pada perilaku yang dimiliki oleh sistem, diantaranya kebutuhan perangkat lunak, perangkat keras, serta *user* sebagai bahan analisis kekurangan dan kebutuhan yang harus di penuhi dalam perancangan sistem yang akan diterapkan.

1) Analisis Perangkat Lunak (*Software*)

Perangkat lunak digunakan dalam sebuah sistem yang merupakan sebuah kumpulan dari perangkat yang mendukung dalam membangun aplikasi ARScan Objek ini. Perangkat lunak tersebut adalah sebagai berikut:

a) Windows 11

b) Unity

c) Vuforia SDK

d) Photoshop

e) Android Jelly bean

f) Android kit kat

2) Analisis Perangkat Keras (*Hardware*)

Laptop komputer *smartphone* dan *webcam* adalah sebuah perangkat keras yang tidak luput dari perangkat lunak sebagai interaksinya. Perangkat lunak memberikan sebuah perintah-perintah terhadap perangkat keras agar dapat berjalan dengan baik. Dalam pembuatan aplikasi AR Scan Objek ini menggunakan perangkat keras sebagai pendukungnya adalah sebagai berikut:

a) Laptop Asus A46C Core i5 VGA nVidia dengan spesifikasi sebagai berikut:

1. *Processor Intel Core i5 -3317u @1.7Ghz*
2. *LCD 14,0" LED Slim*

3. *Intel HD Graphic + VGA nVidia Gforce 635m . 2 GB*
4. *Memory ram 4gb*
5. *HardDisk 500gb*

b) Smartphone dengan spesifikasi sebagai berikut:

1. *Android OS, 10*
2. *Chipset Qualcomm APQ8064 Snapdragon S4 Pro*
3. *CPU Quad-core 1.5 GHz Krait*
4. *Camera Primary 13.1 MP, autofocus, LED flash , secondary 2.2 MP, 1080p@30fps, video 1080p@30fps, HDR*
5. *Memory 256 GB, 8 GB RAM*
6. *Resolution 1080 x 1920 pixels (~441 ppi pixel density)*

b. Kebutuhan Fungsional

Kebutuhan fungsional merupakan penjelasan proses fungsi yang berupa penjelasan secara terinci setiap fungsi yang digunakan untuk menyelesaikan masalah. Kebutuhan fungsional berhubungan dengan fitur *software* yang akan dibuat atau dikembangkan. Berikut ini adalah tahapan analisis kebutuhan fungsional Sistem Aplikasi mobile AR Scan Objek. Analisis yang dilakukan dimodelkan dengan menggunakan *UML (Unified Modeling Language).* Tahap-tahap pemodelan dalam analisis tersebut antara lain identifikasi aktor, *Usecase Diagram, Class Diagram, Sequence Diagram,* dan *Activity Diagram.*

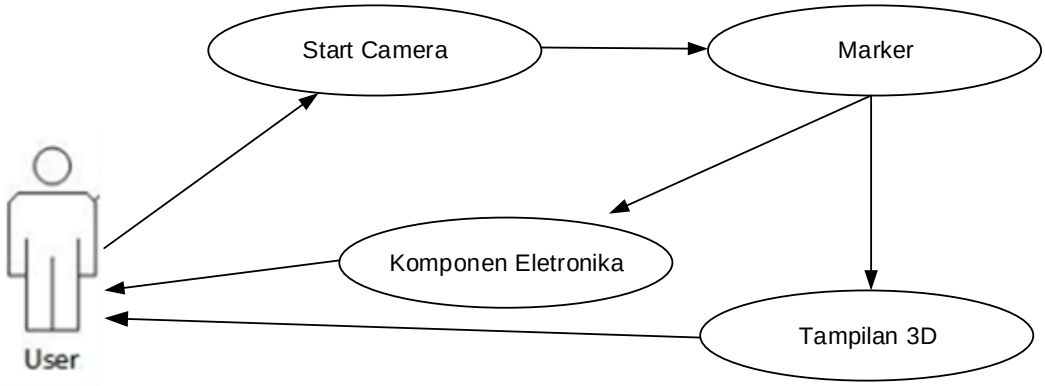
BAB IV

HASIL DAN PEMBAHASAN

A. Rancangan Sistem yang di Usulkan

Sistem yang penulis buat adalah dengan metode pengunjung mengambil alat berupa smartphone berbasis android untuk selanjutnya menggunakannya sebagai Alat untuk memunculkan struktur tumbuhan dalam bentuk 3D dan animasi.

1. *Use Case* Diagram



Gambar 4.1. Diagram *User* Case system yang di ajukan

Penjelasan *Use Case* Diagram di atas :

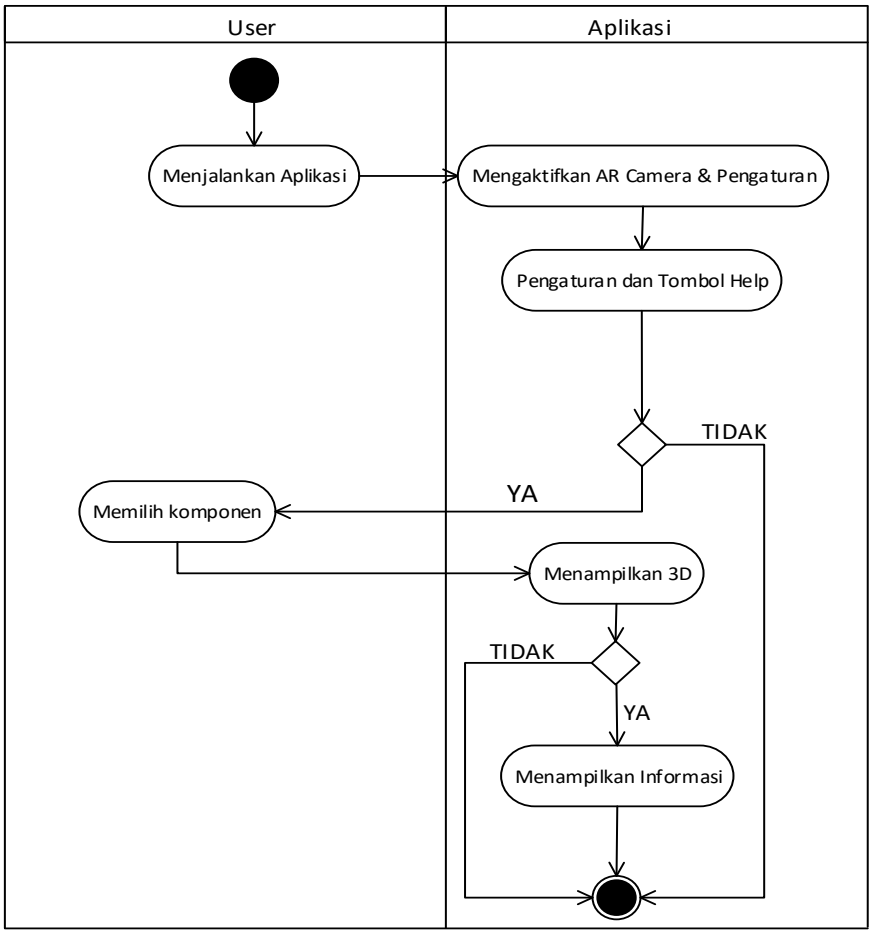
a. *Actor* Pengguna

Tabel 4.1. Tabel *Actor* Pengguna

Nama *Use Case*	Deskripsi *Use Case*
AR Camera	*Use Case* ini menjelaskan bahwa setelah *user* mengaktifkan aplikasi, akan terbuka *Form* AR kamera yang di dalamnya juga terdapat *Form* pengaturan dan menu *User* sudah dapat mengunakannya untuk mendapatkan informasi dengan mengarahkannya ke marker atau objek.
Markerless /Marker	*Use Case* ini menjelaskan tentang proses Pendeteksian objek
Munculkan Informasi dari struktur tumbuhan	*Use Case* ini menjelaskan mengenai informasi yang tampil setelah Proses struktur tumbuhan
Menu play	*Use Case ini* menjelaskan bahwa jika kita menekan tombol play maka akan tampil sebuah proses struktur tumbuhan
home	*Use Case* ini menampilkan cara penggunaan Aplikasi

2. *Activity* Diagram

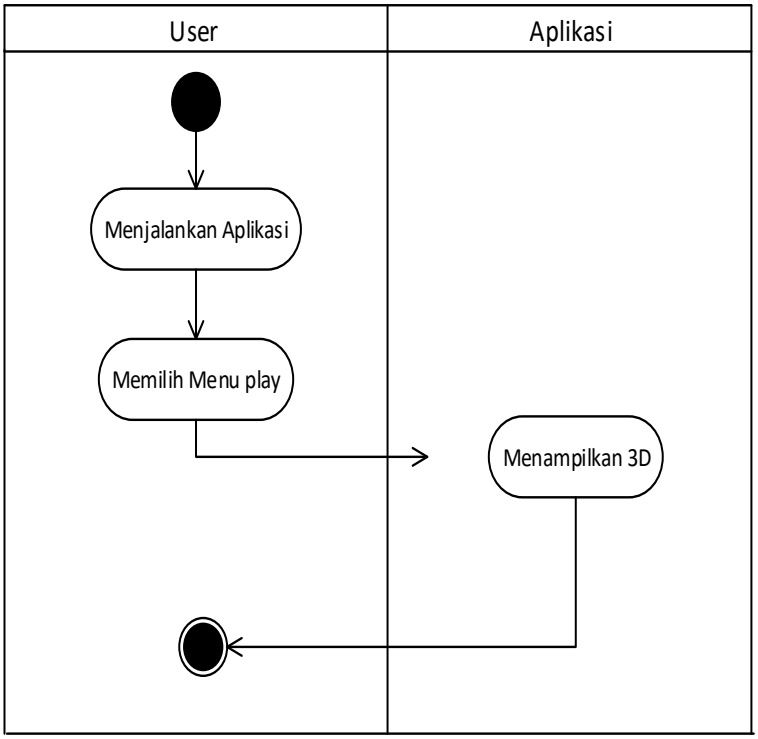
a. *Activity* Diagram Aplikasi



Gambar 4.2. Diagram *Activity* Aplikasi Sistem yang di ajukan

Activity Diagram di atas menjelaskan bahwa ketika *user* menjalankan Aplikasi ini akan langsung di arahkan ke AR kamera yang di dalamnya juga sudah tersedia menu pengaturan dan tombol *Help*, apabila *user* memilih tumbuhan akan menampilkan informasi komponen eletronika dalam bentuk 3D, jika terkendala koneksi Aplikasi akan keluar, dan jika marker tidak terdeteksi dengan baik, informasi yang di inginkan tidak akan tampil.

b. Actifity Diagram Menu play

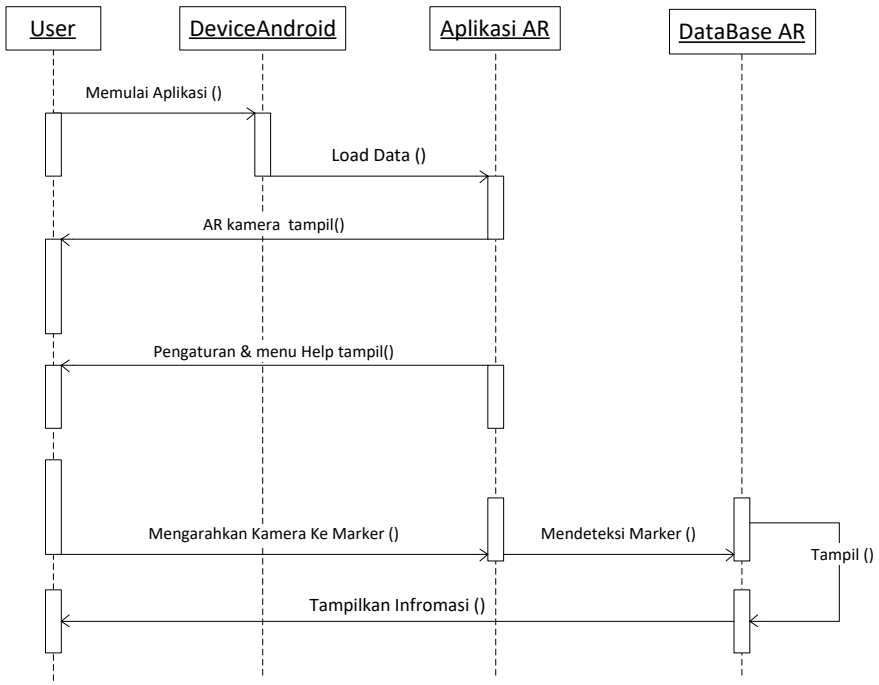


Gambar 4.3. Diagram *Activity Menu play* Sistem yang di Ajukan

Aktifity Diagram di atas menjelaskan bahwa jika *user* menjalankan Aplikasi ini di bagian sudut atas aplikasi terdapat tombol play yang apabila di tekan akan tampil sebuah *PopUp* yang berisikan proses Memilih komponen yang diinginkan.

3. *Sequence* Diagram

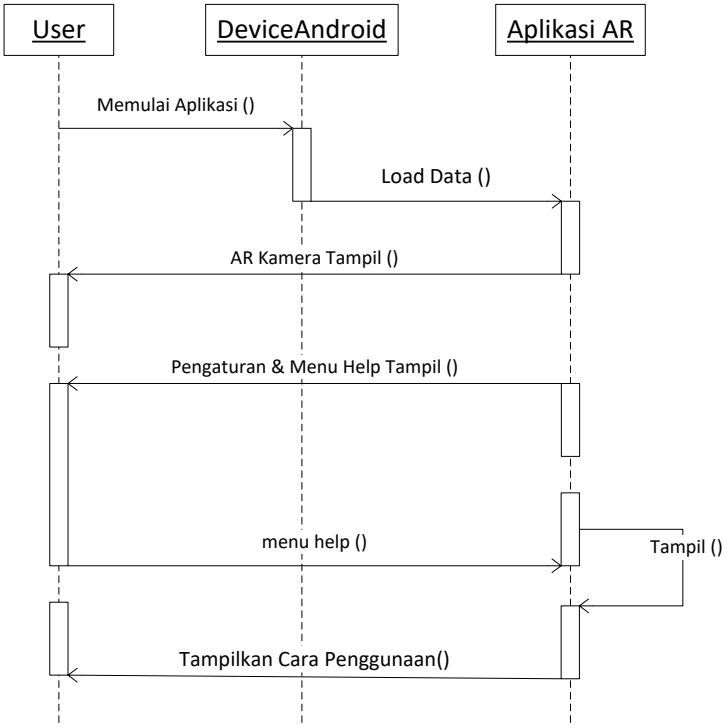
a. *Sequence* Diagram Aplikasi



Gambar 4.4. Diagram *Sequence* Aplikasi Sistem yang di ajukan

Diagram *Sequence* di atas menjelaskan bahwa ketika *user* memulai Aplikasi maka device Android yang di gunakan akan meload data untuk selanjutnya memulai aplikasi ini, setelah itu akan tampil AR camera dan Menu pengaturan serta tombol, setelah AR kamera tampil *user* akan mengarahkan device android tersebut ke *Markeless* yang telah di sediakan yang kemudian aplikasi melakukan proses pendeteksian *Markeless* untuk selanjutnya melakukan pencocokan dengan database jika sesuai dengan data base informasi dari objek akan tampil komponen eletronika dalam bentuk 3D.

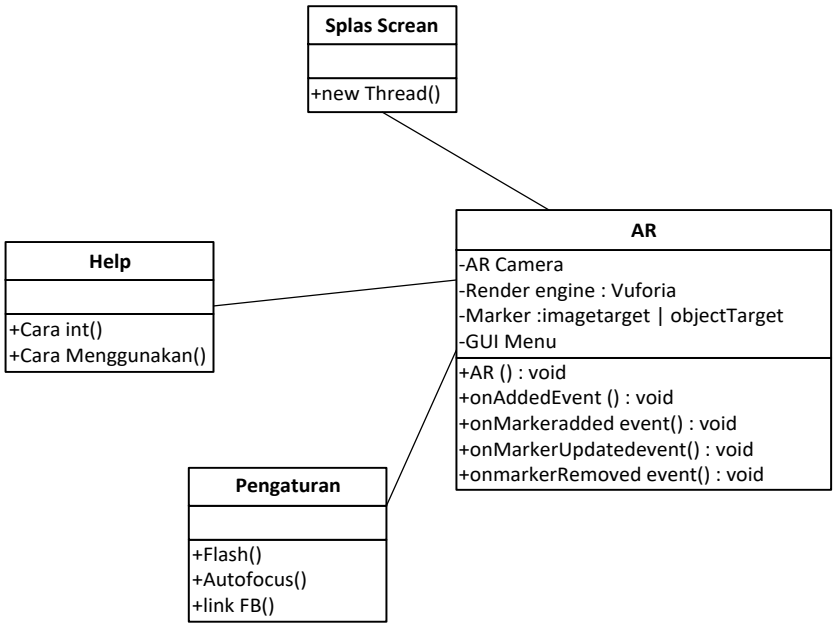
b. *Sequence* Diagram Menu *Help*



Gambar 4.5. Diagram Menu *Help* Sistem yang di ajukan

Diagram *Sequence* di atas menjelaskan bahwa ketika *user* menjalankan Aplikasi akan tampil AR kamera di sertai dengan menuPengaturan dan tombol *Help*, ketika *user* menekan tombol *Help* Aplikasi Akan menampilkan *PopUp* berupa Cara penggunaan dari Aplikasi ini.

4. *Class* Diagram



Gambar 4.6. Diagram *Class* Sistem yang di ajukan

Class Diagram di atas menjelaskan keterkaitan antara setiap *Form* dan fungsi-fungsi yang ada di dalamnya, seperti ketika Aplikasi di jalankan pertama kali maka yang akan tampil pertama kali adalah munculnya sebuah splashcren sebelum masuk ke *Form* AR yang merupakan inti dari aplikasi ini di dalamnya proses menampilkan komponen eletronika dalam bentuk 3d terjadi, adapun jika kita menekan tombol *Help* maka akan di arahkan ke *Form PopUp* yang berisi Cara penggunaan aplikasi ini secara umum.

B. Detail Aplikasi

1. *Form* AR Kamera



Gambar 4.7 *Form* AR Kamera

Gambar di atas adalah tampilan AR kamera ketika *user* telah mengaktifkan Aplikasi ini. nantinya *user* sudah dapat memilih mulai AR maka aplikasi ini akan menampilkan aktif kamera untuk mendapatkan Informasi renang di dalam AR kamera ini juga di sertakan tombol play dan *Form home* dimana di dalamnya terdapat beberapa tombol yang memiliki fungsi-fungsi tertentu.

2. *Form* panduan



Gambar 4.8. Menu panduan

Gambar tampilan di atas adalah sebuah *PopUp* informasi yang tampil apabila *user* menekan tombol panduan pada Menu utama, isinya kurang lebih cara penggunaan Aplikasi secara umum.

3. *Form* Tentang



Gambar 4.9. Menu tentang

Gambar tampilan di atas adalah sebuah *PopUp* informasi yang tampil apabila *user* menekan tombol tentang pada Menu utama, isinya kurang lebih cara penggunaan Aplikasi secara umum.

4. *Form* Mulai AR



Gambar 4.10. Menu Mulai AR

Gambar tampilan di atas adalah sebuah *PopUp* informasi yang tampil apabila *user* menekan tombol Mulai AR pada Menu utama, isinya kurang lebih cara penggunaan Aplikasi secara umum.

5. *Form* resistor



Gambar 4.11. Menu resistor

Gambar tampilan di atas adalah sebuah *PopUp* informasi yang tampil apabila *user* menekan tombol resistor maka akan muncul 3D Resistor pada Menu utama, isinya kurang lebih cara penggunaan Aplikasi secara umum.

6. *Form* Kapasitor



Gambar 4.12. Menu Kapasitor

Gambar tampilan di atas adalah sebuah *PopUp* informasi yang tampil apabila *user* menekan tombol kapasitor maka muncul 3d kapasitor pada Menu utama, isinya kurang lebih cara penggunaan Aplikasi secara umum.

7. *Form* Informasi

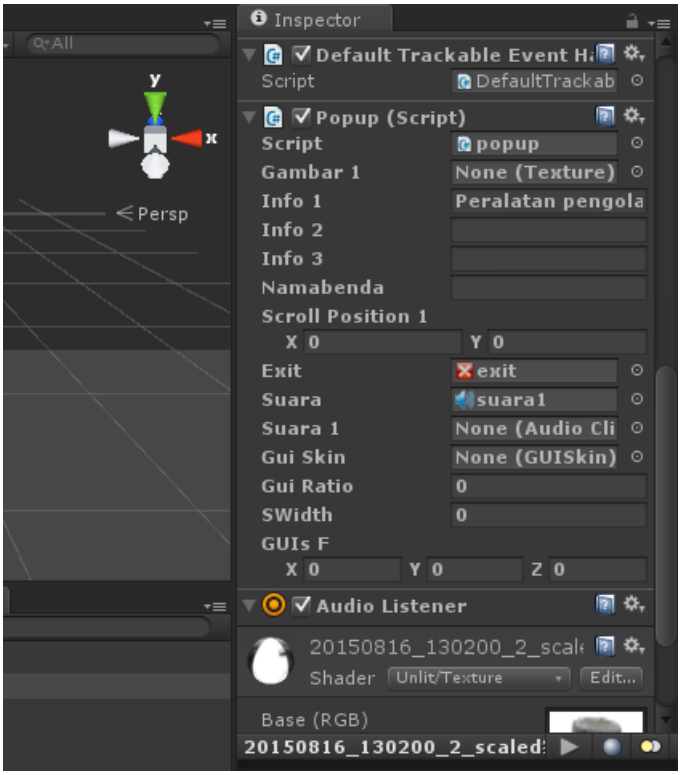


Gambar 4.13. Menu informasi

Gambar tampilan di atas adalah sebuah *PopUp* informasi yang tampil apabila *user* menekan tombol informasi maka muncul informasi komponen eletronika pada Menu utama, isinya kurang lebih cara penggunaan Aplikasi secara umum.

C. Rancangan Input dan Output

1. Rancangan Input



Gambar 4.14. Input Data Aplikasi

D. Pengujian Sistem

1. Implementasi

Implementasi sistem merupakan tahap penerapan dari suatu teknologi yang didesain untuk siap dioperasikan. Pada tahap ini design sistem yang telah dirancang

diterapkan ke dalam bahasa pemrograman yang kemudian di lakukan pengujian sistem.

Adapun bahasa pemrograman yang digunakan untuk membuat aplikasi ini adalah bahasa pemrograman C# (Sharp).

a. Kebutuhan *Hardware*

Spesifikasi minimum *hardware* pengguna aplikasi ini adalah sebagai berikut :

Tabel 4.2. Spesifikasi *Hardware Smartphone*

Jenis	Spesifikasi
Smartphone	Semua Merk
Processor	Intel® pentium™ 1.60 Ghz
RAM	4 GB
Memori	6 GB
Camera	5 MP (Autofokus : Ya || Flash : Ya)
Display	Touchscreen 4.0 inches

b. Kebutuhan *Software*

Spesifikasi minimum *software* pengguna aplikasi ini yaitu sebagai berikut :

Tabel 4.3. Spesifikasi *Software*

Jenis	Spesifikasi
Sistem Operasi	Android Ver. 11

2. Pengujian aplikasi

a. Metode Pengujian

Pengujian aplikasi dilakukan dengan menggunakan dua metode pengujian yaitu Pengujian *Black Box (White box testing)* dan Pengujian *White Box (Black Box testing).*

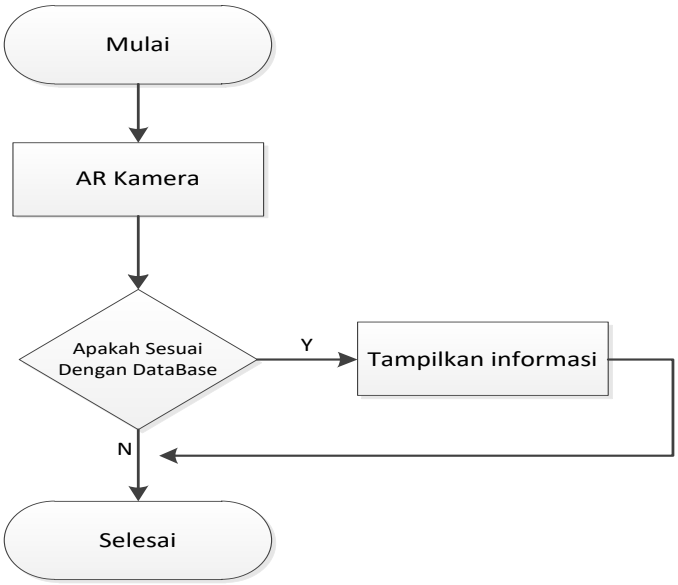
b. Teknik Pengujian

Pengujian *black-box* berkaitan dengan pengujian yang dilakukan pada *interface* perangkat lunak. Meskipun didesain untuk mengungkapkan kesalahan, pengujian *black-box* digunakan untuk memperlihatkan bahwa fungsi-fungsi perangkat lunak adalah operasional, bahwa input diterima dengan baik dan output dihasilkan dengan tepat, dan integritas informasi external (seperti file data) dipelihara. Pengujian blackbox menguji beberapa aspek dasar suatu aplikasi dengan sedikit memperhatikan struktur logika internal perangkat lunak tersebut. sedangkan Pengujian *WhiteBox* berfokus pada efektifitas aplikasi yang dirancang

c. Pengujian *White Box*

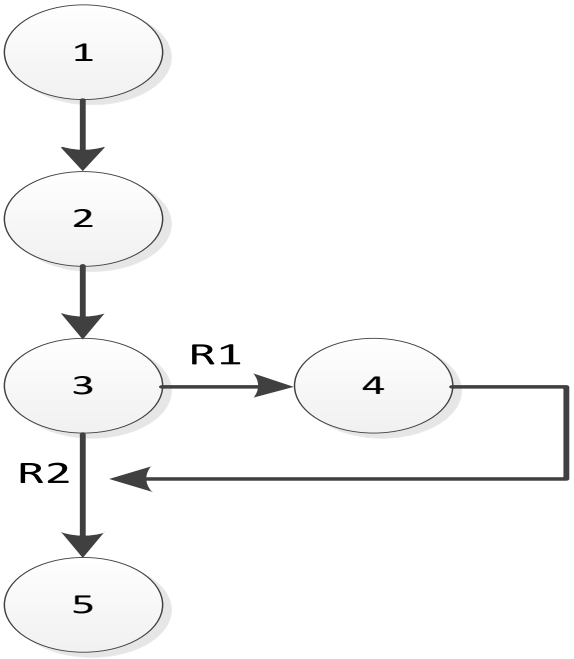
1. *White Box* Aplikasi

    a. *Flowchart* Aplikasi



Gambar 4.15. *Flowchart* Aplikasi

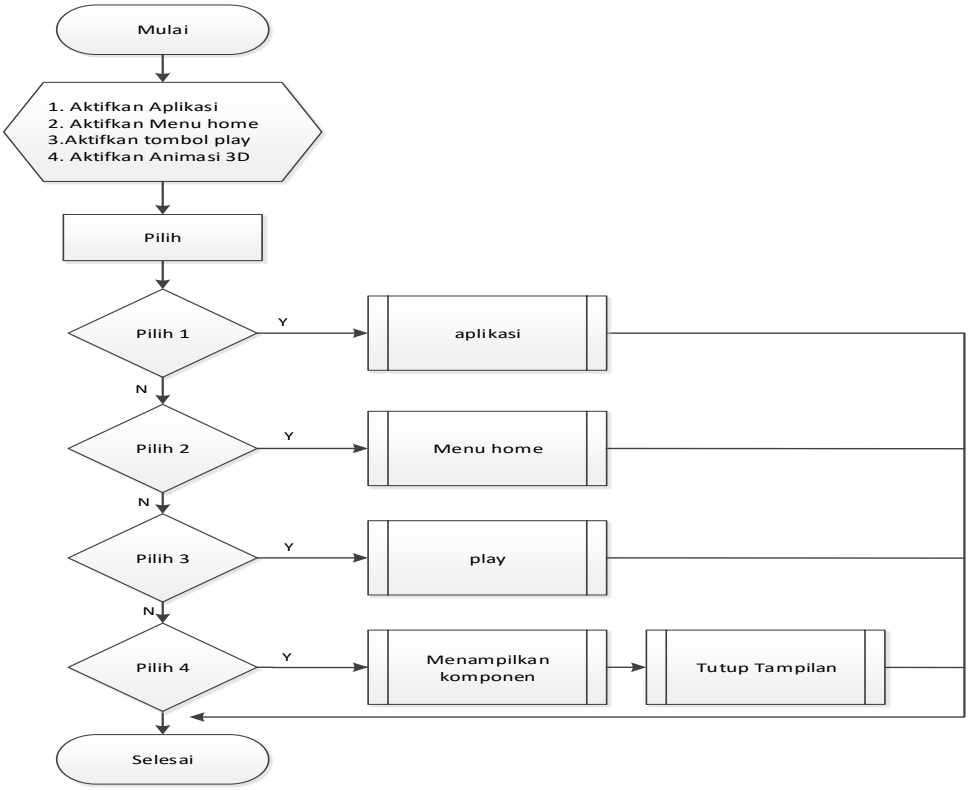
b. *Flowchart* Pengaturan Aplikasi



Gambar 4.16. *Flowchart* Pengaturan Aplikasi

c. *FlowGraph* Aplikasi

Dari *Flowchart* Aplikasi yang di gunakan untuk pengujian perangkat lunak, maka di tentukan *FlowGraph* sebagai berikut :



Gambar 4.17. *FlowGraph* Aplikasi

d. *Proses* Perhitungan

Dari Gambar *FlowGraph* di atas dapat di lakukan proses perhitungan sebagai berikut :

a) Menghitung C*yclomatic Complexcity* V(G) dari *Egde* dan *Node*:

Dengan Rumus : V (G) = E – N + 2

N(*node*) = 5

E(*edge*) = 5

P (*predikat node*) = 1

Penyelesaian : V(G) = E – N + 2

= 5- 5 + 2

= 2

Predikat *Node* (N) = P + 1

= 1 + 1

= 2

b) Berdasarkan perhitungan C*yclomatic Complexcity* dari *FlowGraph* diatas memiliki *Region* = 2

c) *Independent Path* pada *FlowGraph* diatas adalah :

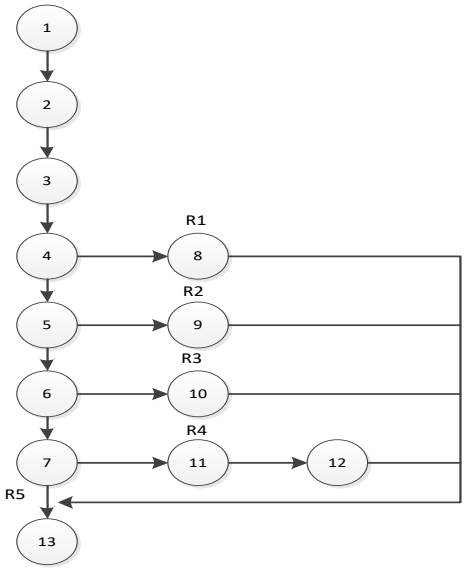
Path 1 = 1 – 2 – 3 – 4 – 5

Path 2 = 1 – 2 – 3 – 5

d) Grafik Matriks

	1	2	3	4	5	E-1
1		1				1-1=0
2			1			1-1=0
3				1	1	2-1=1
4					1	1-1=0
5						
Zum (E+1)						1+1=2

e. *FlowGraph* Pengaturan Aplikasi



Gambar 4.18. *FlowGraph* Pengaturan Aplikasi

f. Proses Perhitungan

Dari Gambar *FlowGraph* di atas dapat di lakukan proses perhitungan sebagai berikut :

a) Menghitung C*yclomatic Complexcity* V(G) dari *Egde* dan *Node*:

Dengan Rumus : V (G) = E – N + 2

N(*node*) = 13

E(*edge*) = 16

P (*predikat node*) = 1

Penyelesaian : V(G) = E – N + 2

= 16- 13 + 2

= 5

Predikat *Node* (N) = P + 1

= 4 + 1

= 5

b) Berdasarkan perhitungan C*yclomatic Complexcity* dari *FlowGraph* diatas memiliki *Region* = 5

c) *Independent Path* pada *FlowGraph* diatas adalah :

Path 1 = 1 – 2 – 3 – 4 – 8 – 13

Path 2 = 1 – 2 – 3 – 4– 5– 9 – 13

Path 3 = 1 – 2 – 3 – 4– 5– 6– 10 – 13

Path 4 = 1 – 2 – 3 – 4– 5– 6– 7 – 11 – 12 – 13

Path 5 = 1 – 2 – 3 – 4– 5– 6 – 7– 13

d) Grafik Matriks

	1	2	3	4	5	6	7	8	9	10	11	12	13	E-1
1		1												1-1=0
2			1											1-1=0
3				1										1-1=0
4					1			1						2-1=1
5						1			1					2-1=1
6							1			1				2-1=1
7											1		1	2-1=1
8													1	1-1=0
9													1	1-1=0
10													1	1-1=0
11												1		1-1=0
12													1	1-1=0
13														
Zum (E+1)														4+1=5

g. Pengujian *Black Box*

Tabel 4.4**.** *Black Box* resistor

Test Factor	Hasil	Keterangan
Jika mendeteksi marker/objek dengan stabil	✓	Berhasil, Karena tampil 3d resistor
ScreenShoot		
		

Tabel 4.5. *Black Box* kapasitor

Test Factor	Hasil	Keterangan
Jika mendeteksi marker/objek dengan stabil	✓	Berhasil, Karena tampil 3d kapasitor
ScreenShoot		
		

Tabel 4.6. *Black Box* induktor

Test Factor	Hasil	Keterangan
Jika mendeteksi marker/objek dengan stabil	✓	Berhasil, Karena tampil induktor
ScreenShoot		



3. Pengujian Jarak

Berdasarkan gambar di atas Aplikasi dapat mengenali marker Eletronika ini dengan baik sama seperti marker sebelumnya harus dengan bantuan pencahayaan yang bagus dan kualitas Kamera pada saat di arahkan.

Adapun Hasil Uji Pengukuran dari Pendeteksian Marker di atas adalah Sebagai berikut :

Tabel 4.6. Hasil Pengukuran Marker

Jarak	Marker	KET
20 cm	✖	✓=terlihat ✖= tidak terlihat
25 cm	✖	
30 cm	✓	
35 cm	✓	

40 cm	✓	
45 cm	✓	
50 cm	✓	
Derajat Kemiringan	**Marker**	
30^0	×	
35^0	×	
40^0	×	
45^0	×	
50^0	×	
55^0	×	
60^0	×	
65^0	×	
70^0	✓	
75^0	✓	
80^0	✓	
85^0	✓	
90^0	✓	

Dari table di atas dapat di ambil kesimpulan sebagai berikut :

1. Resistor
   - Jarak Minimum : 30 cm
   - Jarak Maximum : 50 cm
   - Jarak Optimal : 40 cm
   - Derajat Kemiringan : 70^0
   - Derajat Optimal : 90^0
2. Kapasitor
   - Jarak Minimum : 25 cm
   - Jarak Maximum : 50 cm
   - Jarak Optimal : 30 cm
   - Derajat Kemiringan : 70^0

- Derajat Optimal : 90^0

3. Induktor

- Jarak Minimum : 20 cm
- Jarak Maximum : 35 cm
- Jarak Optimal : 25 cm
- Derajat Kemiringan : 75^0
- Derajat Optimal : 90^0

4. Dioda

- Jarak Minimum : 20 cm
- Jarak Maximum : 35 cm
- Jarak Optimal : 25 cm
- Derajat Kemiringan : 80^0
- Derajat Optimal : 90^0

5. Transistor

- Jarak Minimum : 20 cm
- Jarak Maximum : 35 cm
- Jarak Optimal : 25 cm
- Derajat Kemiringan : 75^0
- Derajat Optimal : 90^0

6. IC

- Jarak Minimum : 20 cm
- Jarak Maximum : 35 cm
- Jarak Optimal : 25 cm

- Derajat Kemiringan : 75^0

  Derajat Optimal : 90^0

7. Saklar

- Jarak Minimum : 20 cm
- Jarak Maximum : 35 cm
- Jarak Optimal : 25 cm
- Derajat Kemiringan : 75^0

  Derajat Optimal : 90^0

Selain dari Marker di atas Penulis juga melakukan percobaan terhadap marker tertentu dengan tujuan untuk menguji sistem dari Aplikasi ini, khususnya dalam hal pendeteksian guna mengetahui apakah dapat di terapkan pada komponen Eletronika nyata (bukan dalam bentuk Marker / gambar).

BAB V

PENUTUP

A. Kesimpulan

Berdasarkan hasil penelitian dan pembahasan yang telah dilakukan oleh penulis maka dapat di tarik kesimpulan sebagai yaitu :

Untuk menampilkan objek *Augmented Reality* bidang *area tracking* harus berjarak jauh dari kamera pengguna dan bidang *area tracking* harus memliki motif/corak warna yang beragam serta memiliki cukup pencahayaan agar cepat dalam proses penampilan Objek 3D.

B. Saran

Saran yang dapat diajukan oleh peneliti mengenai pengembangan adalah sebagai berikut:

1. Media pembelajaran berbasis android pada materi eletronika dan 3d kompopnen menggunakan teknologi *augmented reality* akan lebih baik jika dapat dikembangkan dengan menambah materi lainnya.
2. Menambahkan pilihan bahasa.

DAFTAR PUSTAKA

Arifianto, R. 2017. Materi *Flowchart. From* https://rahmatarifianto.wordpress .com/2014/11/20/pengertian-*Flowchart*-dan-jenis-jenisnya.html. (11 Nov 2021).

Komputer, Wahana (2013), Pengenalan Android, Jakarta, Penerbit:ANDI

Adhitya Wibawa (2015), Pemanfaatan *Augmented Reality* Sebagai Media Marker, Penerbit Andi

Filus, Teo (2017),Pemrograman C++. Jakarta Penerbit: Andi

Filus, Teo. 2017. Pengenalan Pemrograman C#. From https://www.codepolitan. com/pengenalan-bahasa-pemrograman-c-587effa1cb95b (25 Nov 2021).

Ismayani, Ani. 2020. Membuat Sendiri Aplikasi Augmented Reality. Jakarta Pusat : Elex Media Komputindo.

Kadir, Abdul. 2013. Teori dan Aplikasi Pengolahan Citra. Yogyakarta: Andi Offset.

Komputer, Wahana. 2013. Step by Step Menjadi Pemrograman Android. Yogyakarta : Andi.

Putra, Adhitya Wibawa. 2015. Apa Itu Vuforia. From https://teknojurnal. com/vuforia/#:~:text=Vuforia%20ini%20sendiri%20merupakan%20SDK,yang%20memiliki%20teknologi%20Augmented%20Reality. (25 Oktober 2021)

Zaky, Ali. 2016. Animasi Karakter Dengan Blender dan Unity. Jakarta Pusat : Elex Media Komputindo.

N, Sora. 2015. Pengertian *UML* dan Jenis - jenisnya. *From http://www.* pengertianku.net/2015/09/pengertian-*uml*-dan-jenis-jenisnya-serta-contoh-diagramnya.*html.* (11 Nov 2021).

Nadira, Sujaini dan Pratiwi. 2016. Implementasi *Augmented Reality* Pada Brosur Teknik Informatika Universitas Tanjungpura Menggunakan Metode *Marker*. Teknik Informatika., vol. 5, no. 1, pp. 40–48.

Hermawan dan Hariadi. 2015. Pemanfaatan *Augmented Reality* Sebagai Media Informasi Kampus Menggunakan Brosur. E-Journal Tek. Inform., vol. 13, no. 1, pp. 30–37.

Zaky, Ali (2016), *Unity* merupakan game engine yang ber-multi*platform*. *Unity* mampu di publish menjadi Standalone (.exe), berbasis web, *Android*, IoS Iphone, XBOX, dan PS3.